

논문 2015-52-3-5

소형셀 환경에서 코어망 오프로딩을 위한 캐시 알고리즘

(Caching Algorithm for Core Network Offloading in Smallcell Environment)

정 소 이*, 김 재 현**

(So-Yi Jung and Jae-Hyun Kim[Ⓢ])

요 약

본 논문에서는 소형셀 환경에서 사용자의 context를 반영한 local caching 알고리즘을 제안한다. 소형셀 내부에 캐시를 이용하면 코어망으로 전달되는 트래픽을 감소시킬 뿐만 아니라 네트워크 비용을 절감시키고 통신의 성능을 증가시킬 수 있다. 제안하는 알고리즘은 소형셀 특성을 활용해 각 소형셀 사용자에게 맞는 적절한 데이터를 캐시에 저장한다. 이를 통해 제한된 캐시 저장 공간을 효율적으로 사용하고 고전적인 웹 캐시 방식에 비해 캐시 적중률을 높였다. 성능분석을 위해 코어망 트래픽 감소량을 나타내는 cache efficiency를 정의하고 제안한 알고리즘의 성능을 검증한 결과 기존 웹 캐시 방식에 비해 200%의 성능향상이 있었다.

Abstract

In this paper, we propose a smallcell local caching algorithm under user's context in smallcell environment. The proposed system reduces traffic to core network and the network cost, but increases its performance. The proposed algorithm precache suitable files using smallcell's regional characteristics and target's personality. It can adjust a storage allocation to make effective usage of our limited cache storage capacity. In order to evaluate the performance of the proposed cache algorithm, we define the cache efficiency, the decrement of core network traffic. The simulation results show that the proposed algorithm can improve performance by about 200% compared to existing web cache scheme.

Keywords : Smallcell, Offloading, Local Caching

I. 서 론

LTE(Long Term Evolution) 서비스의 급속한 확산에 따라 이동전화 가입자 수가 꾸준히 증가하고 있다.

2014년 9월 기준 국내 3사 이동통신사의 LTE 가입자 수는 3천 4백만 명을 넘어섰다^[1]. 고도화 된 망을 통해 과거 속도와 용량의 한계로는 불가능했던 실시간 스트리밍, 고화질 동영상 재생 등 대용량 서비스를 보다 편리하게 이용하게 되었다. 그 결과 LTE 가입자는 3G 가입자보다 3배 많은 데이터를 사용하고 있다^[2~3].

사용자의 트래픽 요구 사항과 사용 용량이 급증함에 따라 한정된 주파수 자원으로 많은 용량을 처리하기 위한 다양한 기술 개발이 진행 중이다. 5G 무선통신 네트워크에서는 4G 무선통신 네트워크보다 높은 통신 용량, 사용자 측면에서 향상된 QoE(Quality of Experience) 성능, 향상된 에너지 효율 등을 요구한다. 이러한 요구

* 학생회원, ** 정회원, 아주대학교 전자공학과
(Department of Electrical and Computer Engineering, Ajou University)

Ⓢ Corresponding Author(E-mail: jkim@ajou.ac.kr)

※ 본 연구는 미래창조과학부의 정부출연금사업의 일환으로 수행하였음.[15Z1110, 트래픽 용량 증대를 위한 액세스 네트워크 원천기술 연구]

Received ; December 12, 2014 Revised ; January 9, 2015

Accepted ; March 4, 2015

사항을 만족시키기 위해 다양한 기술 논의가 활발히 이루어지고 있다^[4]. 특히 코어망의 과부하를 방지하고 고품질의 서비스를 제공하기 위해 HetNet(Heterogeneous Network)에 대한 관심이 높아지면서 HetNet의 핵심 구성 요소로서 소형셀 역할이 증대되고 있다. 매크로셀로 운영되어 오던 기존의 네트워크 문제를 해결하는 방법으로 매크로 셀 내에 소형셀을 혼용하는 방식을 채택하였다.

현재 대부분의 소형셀은 주거용 펌토셀로 구성되어 있으며 미래에는 피코셀, 마이크로셀 등과 같은 다양한 종류의 소형셀 수가 증가할 것으로 전망하고 있다^[5]. 전세계적으로 소형셀을 도입하는 사업자 수가 빠르게 증가하고 있어 앞으로 소형셀은 소형셀 기지국 역할 뿐만 아니라 가정에서 인터넷 서비스의 중심이 될 것으로 예상된다.

모바일 트래픽의 급증으로 사업자의 부담을 덜기 위해 코어망을 사용하지 않고 공용망으로 연결되는 데이터 오프로딩의 필요성이 증대되고 있다. 모바일 기기에서 Wifi 및 소형셀 네트워크를 통해 오프로딩되는 모바일 트래픽은 계속 증가할 것으로 전망하고 있으며, 2018년에 이르면 셀룰러 망에서 생성되는 트래픽을 넘어설 것으로 예상하고 있다^[6]. 앞으로 소형셀은 매크로셀을 보완하여 추가적인 용량을 제공하는 기존 개념에서 발전하여 가정과 사무실에 독자적으로 용량을 제공하여 모바일 네트워크의 핵심 요소로 자리매김 할 것으로 예측된다.

이렇게 급증하는 소형셀 환경에서 폭발적으로 증대되는 트래픽으로 인한 네트워크 부하를 감소시키는 하나의 방안으로 소형셀 기반 캐시 알고리즘은 다양하게 연구되어 왔다. Ahlehagh의 연구에서는 사용자의 preference를 토대로 가장 요청이 없는 비디오를 캐시에서 삭제하거나 가장 많은 요청이 예상되는 비디오를 미리 저장하는 알고리즘을 제안하였다^[7]. Baştuğ의 연구에서는 요청된 파일들의 popularity 통계정보에 근거해서 캐시를 결정하는 기법을 제안하였다^[8]. Blasco의 연구에서는 소형셀에 캐시함으로서 얻을 수 있는 보상 값을 통해 캐시에 저장 할 파일을 선택하는 알고리즘을 제안하였다^[9]. 하지만 기존 연구들은 소형셀 사용자 성격이 아닌 일반적인 사용자의 특성에 의존하고 있다. 또한 비디오나, 파일이라는 일관적인 데이터 크기를 전제하고 있어 캐시 알고리즘의 현실성이 저하되고 있다.

따라서 본 논문에서는 소형셀과 소형셀 사용자의 특징을 반영하여 기존의 연구와 차별성을 둔 local caching 알고리즘을 개발하고자 한다. 또한 알고리즘의 현실성을 강화하기 위해 실제 소형셀 사용자의 검색어 행태를 분석하고 실제 발생하는 트래픽을 계산하여 시뮬레이션에 적용하고자 한다.

II. 기존 연구

1. 웹 캐시 교체 알고리즘

캐시에는 저장할 수 있는 공간이 한정되어 있기 때문에 캐시 서버의 공간이 부족하게 되면 효율을 높이기 위해 캐시 내 문서를 제거해야 한다. 이때 사용되는 캐시 교체 정책은 성능에 많은 영향을 주게 된다.

LRU(Least Recently Used) 알고리즘은 사용자의 최근 웹 객체 빈도 이력을 이용하여 캐시에 저장되어 있는 객체 중 가장 적게 이용된 객체를 캐시에서 먼저 삭제하는 방식이다^[10]. LFU(Least Frequently Used) 알고리즘은 사용 빈도가 가장 적은 객체를 캐시에서 삭제하는 방식이다^[11]. LRU와 LFU는 웹 캐시 분야에서 전통적인 교체 알고리즘으로 단순하고 견고하기 때문에 가장 널리 사용되고 있다.

2. 소형셀 캐시 연구

소형셀 포럼에서는 소형셀의 목적에 따라 home, enterprise, urban, rural 등으로 구분하고 문서를 발간하고 있다. 특히 home, enterprise 기반 소형셀 사용자들은 같은 목적을 지니고 있거나 성향이 비슷할 확률이 높으며 근접 소형셀 사용자 간의 유사도가 높다. 이러한 특징을 이용해 다양한 소형셀 캐시 방안을 제안하고 있다^[12].

커피숍 체인점과 같은 형태에서 물리적으로는 분산되어 있지만 접근하는 트래픽에 높은 유사도를 보이는 사용자들을 관리하기 위한 분산적 소형셀 캐시 저장 시나리오가 있다. 커피숍 체인점이나 사업자가 POP(Point of Presence) 역할을 하여 고객들이 선호하는 특정 스트리밍 콘텐츠를 제공하고 광고를 통한 수익을 창출한다. 회사 규모에서 소형셀 구축을 원하는 경우는 기업적 소형셀 캐시 저장 시나리오를 사용한다. 이 경우 해당 기업 사원들만 접근 가능한 폐쇄형 모드로 동작하게 된다.

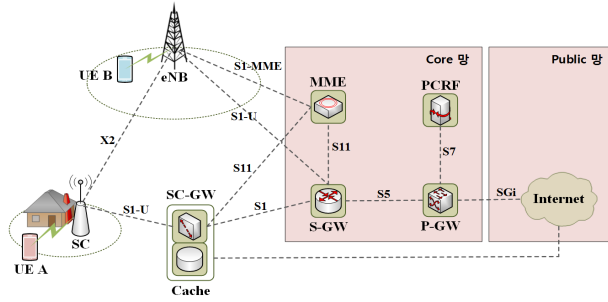


그림 1. 알고리즘을 적용한 LTE 네트워크 구조
Fig. 1. LTE network architecture for proposed algorithm.

III. 제안하는 알고리즘

1. 제안하는 알고리즘을 적용한 LTE 네트워크 구조

LTE 네트워크 참조 모델에 기반하여 소형셀 local caching 알고리즘이 적용될 경우 예상되는 네트워크 구조는 그림 1과 같다.

기존 LTE 네트워크 모델에서 SC-GW(Smallcell Gateway)와 Cache가 추가된다. SC-GW는 캐시 기능이 있는 소형셀이 배치되는 장소에 함께 존재하며 단말의 이동성과 서비스를 관리한다.

제안하는 알고리즘이 적용될 경우 가능한 시나리오는 크게 두 가지가 있다. 단말 A가 요청하는 트래픽이 소형셀 캐시에 존재하지 않을 경우 기존의 방식으로 코어망으로부터 페이지를 불러와야 한다. 다운로드 과정은 경로 1과 같이 복잡한 절차가 된다.

(경로 1) Internet→P-GW→S-GW→SC-GW→UE A

만일 단말 A가 요청하는 트래픽이 소형셀 캐시에 존재한다면 경로 2와 같이 절차가 간략해진다.

(경로 2) SC-GW→UE A

소형셀의 특성상 물리적으로 인접한 소형셀 사용자들은 성향이 유사할 확률이 매우 높다. 이러한 특징은 두 eNB간 X2 인터페이스를 이용하여 트래픽 효율을 높일 수 있다. X2 인터페이스는 서로 인접한 기지국 간 정보를 주고 받을 수 있고 EPC(Evolved Packet Core) 노드의 개입 없이 핸드오버 수행이 가능한 장점이 있다. 이를 이용하여 단말 B가 요청하는 트래픽이 소형셀 캐시에 존재한다면 경로 3과 같이 기존 방식보다 짧은 경로를 이용할 수 있다.

(경로 3) SC-GW→UE B

표 1. 의사코드에 적용된 파라미터

Table 1. Parameter for pseudo code.

Parameter	Value
i	1위부터 100위까지 순위(위)
t	9시부터 22시까지 시간(시)
$X = a, b, c$	A=실시간 검색어, B=타겟 별 검색어, C=지역 별 검색어
$K_{X,t}$	검색어 키워드
$F_{i,t}^X$	파일 크기(Kbytes)
C	전체 캐시 저장 가능 용량(Kbytes)
$C_{X,t}$	t 시각 캐시에 저장된 키워드
$N_{X,t}(x)$	t 시각 캐시에 저장된 키워드 개수(개)
$N_{X,t,1}(x,y)$	최근 일정기간 동안 키워드를 캐시와 비교하여 재 사용률이 0인 키워드 개수(개)
$K_{X,t,0}(x,y)$	$N_{X,t,1}(x,y)$ 에 해당하는 키워드

그림 2. 소형셀 local caching 알고리즘 의사코드
Fig. 2. Proposed algorithm for smallcell local caching.

```

Phase 1 : Choose smallcell information base on keyword.
target user={single man, single woman, office worker,
investor, housewife, university student, teenager}
location={specify area until borough unit}
Phase 2 : Allocate cache capacity each keyword.
At time t : each small cell tracks  $K_{a,t}, K_{b,t}, K_{c,t}$ 
Calculate  $\sum \sum F_{i,t}^a, \sum \sum F_{i,t}^b, \sum \sum F_{i,t}^c$  during
 $i = [1,10], t = [9,22]$ 
Calculate  $T = \sum \sum F_{i,t}^a + \sum \sum F_{i,t}^b + \sum \sum F_{i,t}^c$ 
Find  $\alpha = \sum \sum F_{i,t}^a / T, \beta = \sum \sum F_{i,t}^b / T, \gamma = \sum \sum F_{i,t}^c / T$ 
Phase 3 : Initial cache and update management
Save  $\sum K_{a,t}$  at  $C_{a,t}$  during  $t = [9,22]$ 
If ( $C_{a,t} \geq C \cdot \alpha$ )
  If ( $t \leq 12$ ) then  $k = t - 9$  else  $k = 4$  end if
  Compare  $x = C_{a,t}$  with  $y = \sum_{t=t-k}^t K_{a,t}$ 
  If ( $N_{a,t}(x) - N_{a,t,0}(x,y) > C \cdot \alpha$ ) then
    While ( $N_{a,t}(x) - N_{a,t,0}(x,y) \leq C \cdot \alpha$ )
       $t - k = t - k + 1$ 
       $C_{a,t} = C_{a,t} - K_{a,t,0}(x,y)$  until  $N_{a,t}(x) = C \cdot \alpha$ 
      reset  $y = \sum_{t=t-k}^t K_{a,t}$ 
    end
  else
    While ( $N_{a,t}(x) = C \cdot \alpha$ )
       $C_{a,t} = C_{a,t} + K_{a,t,0}(x,y)$  until  $N_{a,t}(x) = C \cdot \alpha$ 
    end
  end if
else
  Accumulate  $C_{a,t}$ 
End if
Repeat  $K_{b,t}, K_{c,t}$ 

```

2. 제안하는 소형셀 local caching 알고리즘

제안하는 소형셀 local caching 알고리즘은 소형셀 사용자들의 유사한 성향을 이용하여 적절한 트래픽을 캐시에 저장하고자 한다. 구체적으로 소형셀 사용자 집단의 특성과 소형셀의 지역적인 특성을 이용한다. 제안하는 알고리즘의 목적은 상대적으로 규모가 작은 소형셀에 접속한 사용자들의 성향을 충분히 반영하여 요청될 트래픽을 예측하고 캐시에 저장하여 코어망의 부하를 감소시키고 최대의 효율을 만들어 내하고자 하는 것이다. 이를 위한 수단으로 포털 사이트의 검색어를 이용한 알고리즘을 제안한다. 본 논문에서는 국내 주요 검색 포털 사이트 중 전체 시장의 73%의 점유율을 차지하고 있는 네이버의 검색어를 이용하고자 한다. 네이버가 제공하는 검색 서비스 중 일반적으로 모든 사람이 관심을 갖는 실시간 인기 검색어와 소형셀 사용자의 특성이 반영될 수 있는 타겟 별 인기 검색어, 소형셀이 위치한 장소의 특성이 반영될 수 있는 지역 별 인기 검색어를 활용하였다^[13].

그림 2는 제안하는 알고리즘의 의사 코드를 나타내고 사용된 파라미터에 대한 설명은 표 1과 같다. 제안하는 알고리즘은 크게 세 단계로 구분할 수 있다. 먼저 소형셀 사용자 그룹을 싱글남, 싱글녀, 직장인, 재테크족, 주부, 대학생, 청소년 중 소형셀 사용자 특성과 가장 유사한 그룹 하나를 선정한다. 이는 사용자의 로그 분석 혹은 과거 검색 기록을 이용해 선별한다. 소형셀이 위치한 장소는 GPS(Global Positioning System) 또는 과거 검색 기록을 이용해 구 단위까지 선정한다.

두 번째 단계에서는 전체 캐시 가용 가능 용량을 할당한다. 소형셀이 일반적인 웹 캐시 크기와 유사하다는 전제하에 최소 용량으로 최대의 효과를 내하고자 각 키워드를 통해 예상되는 파일 크기를 점수화하고 크기에 비례하여 가용 캐시 용량을 할당한다.

마지막 단계에서는 캐시 업데이트 과정을 수행한다. 업데이트 초기 단계에서는 각 검색어 마다 할당된 캐시 사용 가능 용량까지만 인기 페이지를 채워준다. 이후 유지단계에서는 한 시간마다 LRU 방식을 이용하여 업데이트 된 캐시 내용을 교체하여 준다. 할당된 캐시 사용 가능 용량을 초과하게 되는 경우, 캐시에 저장된 키워드와 최근 일정 기간 동안의 키워드를 비교하여 재 사용률이 제로인 키워드만 사용 가능 용량까지 제거한다.

IV. 성능분석 모델링

제안하는 local caching 알고리즘을 사용하였을 경우 기존 대비 코어망 트래픽이 얼마나 감소하였는지 효율 검증을 위한 모델링을 제안한다. 전체 발생하는 트래픽에서 캐시에 저장하기 위해 사용되는 트래픽과 캐시 미스에 의해 코어망으로부터 발생하는 트래픽의 양을 제거하여 cache efficiency를 도출한다.

$$Eff_{cache} = \frac{T_{total} - (T_{save} + T_{off})}{T_{total}} \times 100(\%) \quad (1)$$

수식 1은 제안하는 local caching 알고리즘을 적용하였을 때 전체 트래픽 대비 코어망 오프로딩을 이용한 캐시 효율을 나타내며 다음과 같은 세 가지 구성 요소로 이루어져있다.

$$T_{total} = \sum_{t=9}^{22} \sum_{i=1}^{100} N_i \cdot L_{i,t,X} \quad (2)$$

T_{total} 은 성능 분석 기간 동안 발생하는 전체 트래픽을 나타낸다. N 은 검색 횟수, L 은 파일 크기를 나타낸다. 사용자로부터 100가지 키워드가 검색된다고 가정할 때 요청되는 전체 페이지의 트래픽 양을 계산한다.

$$t_{save} = t(C_{X,t} = C \cdot \sum \sum F_{i,t}^X / T) \quad (3)$$

$$T_{save} = \sum_{t=9}^{t_{save}} \sum_{i=1}^{100} y \cdot L_{i,t,X} + U_{X,t}([t_{save} + 1, 22]) \cdot L_{i,t,X} \quad (4)$$

T_{save} 은 키워드가 캐시에 저장되기 위해 사용되는 트래픽 양으로 초기단계와 유지단계로 나뉜다. 초기단계는 할당된 캐시 용량까지 캐시를 채우는 과정에 해당된다. 유지단계는 한 시간 간격으로 저장된 캐시와 새로운 키워드를 비교하여 캐시를 업데이트하면서 추가로 키워드를 저장하는 단계이다.

$$U_{X,t} = \begin{cases} 0 = no\ change \\ 1 = replacement \end{cases} \quad (5)$$

$U_{X,t}$ 은 유지단계에서 한 시간 전 키워드와 비교하여 변경된 내용을 나타내는 함수이다. 0인 상태는 캐시 내용이 변하지 않으며 유지되는 상태이고, 1인 상태는 새로운 키워드가 생성되어 해당 페이지 캐시 업데이트가 필요한 상태에 해당된다.

표 2. 웹 서비스 별 파일 크기
Table 2. File size of web service.

Score	Type of service	File size
1	a one time search, article	500Kbytes
2	multiple time search, article contains several images	1500Kbytes
3	webtoon, page contains high definition images	4Mbytes
4	audio streaming	5Mbytes
5	video streaming	6Mbytes
6	download	30Mbytes

$$T_{off} = \sum_{t=9}^{24} \sum_{i=11}^{100} N_i \cdot L_{i,t,X} + \sum_{t=9}^{24} \sum_{i=1}^{10} (1 - H_{X,t}) \cdot L_{i,t,X} \quad (6)$$

T_{off} 는 캐시 되지 않은 키워드가 요청되어 발생하는 트래픽 양을 나타낸다. 일반적으로 캐시 미스라고 부르는 경우에 해당된다. 제안하는 알고리즘에서는 두 가지 형태의 캐시 미스가 발생할 수 있다. 먼저 캐시의 대상에서 제외되었던 11위부터 100위까지의 키워드가 검색되었을 경우 해당 페이지는 코어망으로부터 요청해야 한다. 1위부터 10위 내의 키워드가 검색되더라도 캐시에 저장되어 있지 않은 경우에는 코어망으로부터 트래픽을 받아와야 한다. H 는 캐시 적중률을 나타낸다.

V. 성능분석 결과

1. 시뮬레이션 환경

소형셀 캐시의 최대 저장 가능 용량은 웹 브라우저에서 사용하는 캐시 사이즈를 적용하였다. 최소의 용량으로 최대의 효과를 내기 위해서 일반적으로 한 사람이 사용하는 웹 브라우저 캐시 사이즈 값인 103Mbytes를 사용하였다. 제한된 캐시 사이즈 내에서 서로 다른 특징을 가진 실시간, 타겟 별, 지역 별 세 종류의 검색어가 모두 저장되기 위해서는 각각 최대 캐시 저장 사이즈를 분배해야 된다. 제안하는 알고리즘에서는 파일 크기를 이용하여 할당 비율을 정하고자 한다.

사용자가 웹 브라우저를 이용하는 경우 일반적으로 트래픽을 사용하는 수준에 따라 웹 페이지, 오디오 스트리밍, 비디오 스트리밍 세 단계로 나누는 방식을 많이 이용한다. 본 알고리즘에서는 웹 페이지를 세 단계로 세분화하여 표 2와 같이 여섯 단계의 웹 서비스를 새롭게 정의하였다. 이를 바탕으로 시뮬레이션 기간 동안 네이버 포털 사이트에서 수집된 검색어의 서비스를

표 3. 시뮬레이션 환경
Table 3. Simulation environment.

Parameter	Setting
Type of smallcell	Home, enterprise smallcell
Number of smallcell	1
Number of users	20 peoples
Cache size	103Mbytes
Replacement policy	LRU
Replacement duration	1 hour
Size of keyword	실시간 검색어(A) : 1122.96kbytes 대학생 검색어(B) : 2700.93kbytes 영통구 검색어(C) : 1484.01kbytes
Generation traffic per person ^[14]	Light user : 10.83 times Heavy user : 108.33 times
Traffic Scenario	5 light users, 15 heavy users
Probability to search keyword ^[15]	1위부터 10위 내 : 56.64% 10위부터 100위 내 : 43.36% 101위 이상 : 0%
Keyword Scenario (KS)	KS 1 : A 33.3%, B 33.3% C 33.3% KS 2 : A 50%, B 25% C 25% KS 3 : A 60%, B 20% C 20% KS 4 : A 20%, B 60% C 20%

유추하여 파일 점수를 부여하였다. 이를 통해 실시간 인기 검색어의 파일 점수는 1.508점, 대학교 인기 검색어는 2.386점, 영통구 인기 검색어는 1.969점이 부여되었다. 각 점수를 파일 크기로 환산하여 캐시 사이즈로 할당하면 21.79Mbytes, 52.41Mbytes, 28.80Mbytes가 할당 된다.

2. 시뮬레이션 결과

제안하는 알고리즘을 적용한 시뮬레이션 구현을 위해 표 3과 같은 환경을 구성하였다. 시뮬레이션 기간 동안 실제 인터넷 사용자들의 검색 기록을 토대로 매일 산출된 100가지의 키워드를 이용하여 키워드 검색 확률을 계산하였다. 10위 내 검색 확률이 높아질수록 캐시를 이용하는 빈도가 높아짐을 의미한다. 또한 실시간 검색어, 타겟 검색어, 지역 검색어 키워드 비율이 다른 4가지 시나리오를 구성하였다.

그림 3은 제안하는 local caching 알고리즘과 FIFO 방식의 하루 평균 캐시적중률을 나타낸다. 검색어는 급변하는 것이 아니라 일정 흐름을 유지하는 변수이기 때문에 FIFO 방식보다 LRU를 기반으로 한 local caching 알고리즘의 캐시 적중률이 더 높게 나왔다. 이러한 결과는 소형셀 사용자들이 유사한 성향을 가지고 있어 저장된 캐시를 다시 사용할 확률이 높음을 의미한다.

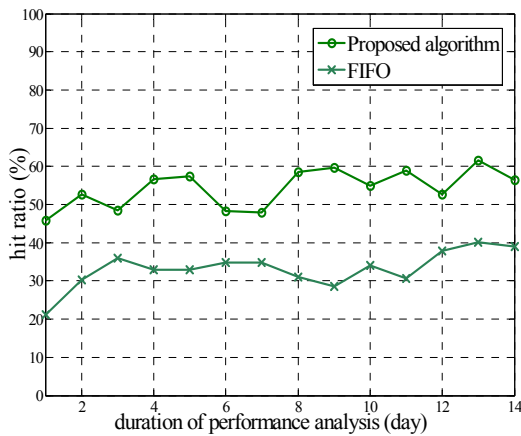


그림 3. 알고리즘 별 날짜에 따른 캐시 적중률
Fig. 3. Cache hit ratio per algorithm.

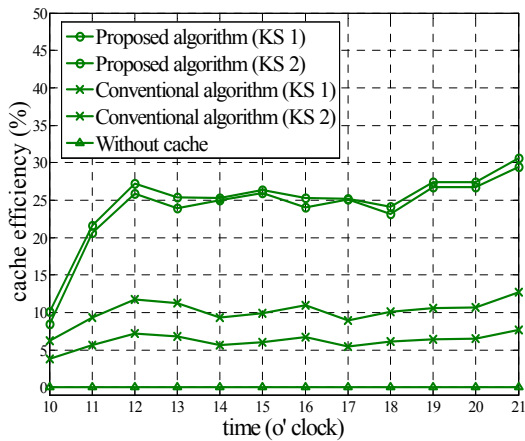


그림 4. 방식 별 시간에 따른 캐시 효율성(KS=1,2)
Fig. 4. Cache efficiency per algorithm(KS=1,2).

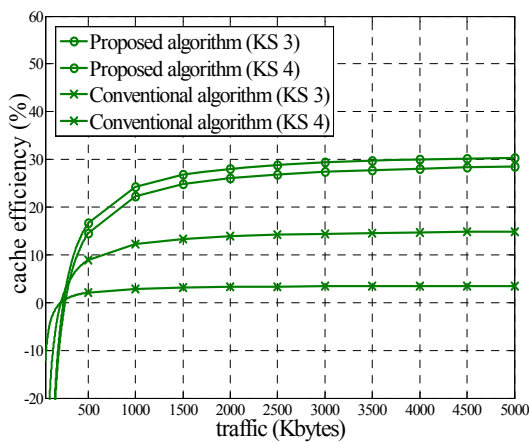


그림 5. 시나리오 별 트래픽에 따른 캐시 효율성 (KS=3,4)
Fig. 5. Cache efficiency per scenario(KS=3,4).

그림 4는 제안하는 알고리즘을 적용하였을 때와, 기존의 웹 캐시 방식, 그리고 캐시가 없을 시 시간에 따른 효율성을 보여준다. 알고리즘을 적용할 경우, 시나리오에 따른 캐시 효율성에 큰 변화가 없다. 반면 알고리즘을 적용하지 않은 기존의 웹 캐시 방식의 경우, 소형셀 특성이 반영되지 않은 실시간 인기 검색어만 의존하기 때문에 그 비율에 따라 효율이 달라진다. 제안하는 알고리즘을 적용하면 작은 캐시 용량에도 충분히 캐시 적중률을 높이고 코어망의 트래픽을 줄일 수 있는 알고리즘의 효율성을 검증하였다.

그림 5는 제안하는 알고리즘을 적용한 경우와 기존의 웹 캐시 방식에서 트래픽 증가에 따른 캐시 효율성을 보여준다. 트래픽이 적게 발생하는 캐시 초기 단계에서는 캐시에 키워드를 저장하기 위한 코어망 사용이 빈번하고 캐시 적중률이 낮아서 캐시 효율성이 마이너스 값을 갖게 된다. 제안하는 알고리즘을 적용하였을 경우 기존의 방식에 비해 캐시 효율성이 빠르게 증가하고 650Kbytes에서 20%를 넘어선다. 이후 발생 트래픽이 증가함에 따라 캐시 효율성이 30%에 수렴하게 된다. 같은 조건에서 웹 캐시 방식은 절반 수준인 15%의 효율을 갖게 된다. 특히 트래픽 발생량이 낮을 경우 효율이 현저히 떨어지게 되며 KS 4 기반에서는 5%의 효율도 달성하지 못한다. 본 논문에서 제안하는 알고리즘은 사용자가 일정 이상의 트래픽이 발생하는 소형셀에 적용하였을 경우 최대의 효율이 발생됨을 확인하였다.

VI. 결 론

본 논문에서는 LTE 시스템에서 소형셀 사용자의 특성을 반영한 local caching 알고리즘을 제안하였다. 캐시 효율성 모델을 이용하여 코어망의 트래픽이 감소하여 성능이 개선된 것을 검증하였고, 시뮬레이션 결과 기존 웹 캐시 방식에 비해 200%의 효율이 증가함을 확인하였다. 본 연구에서 제안하는 local caching 알고리즘은 일반적인 가정이나 사무실 등 일정 수준 이상의 트래픽이 발생하고 캐시 기능을 보유한 소형셀에 탑재될 수 있다. 사용자의 유사도가 높은 집단일수록 코어망으로부터 가져오는 트래픽 양이 감소하여 성능이 큰 폭으로 개선될 것으로 예상된다. 결론적으로 제안하는 알고리즘은 향후 폭발적으로 증대되는 트래픽으로 인한 네트워크 부하를 감소시킬 것으로 기대된다.

REFERENCES

- [1] Ministry of Science, ICT and Future Planning of the Republic of Korea, Telecommunications Policy Bureau, "Subscriber Statistics of Wireless Communication," Sep. 2014.
- [2] Ministry of Science, ICT and Future Planning of the Republic of Korea, Telecommunications Policy Bureau, "Traffic Statistics of Wireless Communication," Sep. 2014.
- [3] S. H. Moon, H. M. Go, J. H. Park and J. S. Choi, "Core Technology and Business Operator Requirements for Next Generation Network Evolution," Journal of the Institute of Electronics and Information Engineers, pp. 26-33, vol. 40, no. 4, April 2013.
- [4] C. H. Lee, J. S. Kim and J. H. Kim, "Future of B4G/5G Wireless Communication Network," Journal of the Institute of Electronics and Information Engineers, pp. 88-98, vol. 40, no. 4, April 2013.
- [5] Small Cell Forum Release Four, SCF 050.04.02, "Market status statistics Q1 2014 - Mobile Experts," June. 2014.
- [6] CISCO, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update 2013 - 2018," Feb. 2014.
- [7] H. Ahlehagh and S. Dey, "Video Caching in Radio Access Network: Impact on Delay and Capacity," in Proc. WCNC 2012, April 2012.
- [8] E. Baştuğ, J. L. Guénégo and M. Debbah, "Proactive Small Cell Networks," in Proc. ICT 2013, May 2013.
- [9] P. Blasco and D. Gündüz, "Learning-Based Optimization of Cache Content in a Small Cell Base Station," in Proc. ICC 2014, June 2014.
- [10] C. Y. Chang, T. McGregor and G. Holmes, "The LRU *WWW proxy cache document replacement algorithm," in Proc. Asia Pacific Web Conference, 1999.
- [11] M. Abrams, C. R. Standridge and G. Abdulla, "Removal Policies in Network Caches for World-Wide Web Objects," in Proc. 1996 ACM Sigcomm Conf., ACM Press, New York, pp. 293-305, 1996.
- [12] Small Cell Forum Release Four, SCF 093.04.01, "Extensions of small cell zone API," June. 2014.
- [13] NAVER, <http://help.naver.com/support/service/>
- [14] NAVER story, <http://story.naver.com/naverbrand/>

— 저 자 소 개 —



정 소 이(학생회원)
2009년~2012년 아주대학교
전자공학과 학사 졸업
2013년~2014년 아주대학교
전자공학과 석사 졸업

<주관심분야 : 소형셀, LTE/LTE-A, 차세대 이동통신 시스템>



김 재 현(평생회원)
1987년~1996년 한양대학교 전산
과 학사 및 석/박사 졸업
1997년~1998년 미국 UCLA 전기
전자과 박사 후 연수
1998년~2003년 Bell Labs, NJ,
USA, 연구원

2003년~현재 아주대학교 전자공학부 교수
<주관심분야 : QoE/QoS, 무선 MAC 프로토콜, IEEE 802.11/15, B5G 통신 시스템, 국방 전술네트워크, 위성시스템 등>